



MOS

P L U S

David Spencer

Extensions and improvement to the
Master 128 Operating System

DABS
PRESS

About MOS Plus

MOS Plus is a ROM specifically designed to plug the gaps in the Master Operating System (Version 3.10), fixing the known bugs and making the Master much easier to use. It isn't a "utility" ROM, containing sector editors, BASIC program fixes, and other routines mainly aimed at programmers. There are many such ROMs on the market, and we didn't see the need to write another. This was borne out of the frustration of using a Master 128 in an office, for normal word processing, spreadsheet and other work, with the usual chore of backing up and renaming files, and all the other things you often do with your machine.

MOS Plus fixes the infamous DFS bug, which writes the wrong length back to an open file when *CLOSE or CLOSE#0, provides format, backup and verify facilities for ADFS discs, and provides a host of minor improvements. Also included is an alarm clock system which can be set to beep at a specified time in the future, or every hour, minute, or second, and a help text shoeing you how to use Edit, VIEW, ViewSheet, Terminal.

The idea is that you install MOS Plus and forget it, knowing that many of the problems often faced using the Master will just disappear. MOS Plus can be used in any ROM socket, and so can be run in the Sideways RAM of a Master (from a disc), or from a ROM cartridge, or, the recommended method, in the single spare internal ROM socket of the Master. Section 2 shows you how to do this.

MOS Plus can be used with Econet, any co-processor, and with virtually any peripheral. Only one restriction exists, the ADFS commands do not work with a SCSI device such as a Winchester drive. You should of course find that formatting a Winchester is a fairly rare occurrence.

Installing MOS Plus

Installing MOS Plus is a fairly simple job for non-technical users and an absolute doddle to anyone who has done it before.

The ROM version

There is one ROM socket inside your Master 128, which can be used to hold programs in this form. To fit the ROM in this socket, ideal if nothing else is already there and you don't want to use a cartridge. Unscrew the four fixing screws on your Master (don't worry it all goes back together again), and lift off the lid. On the right of the circuit board there are three sockets, normally empty which sit in a row from top to bottom. Gently insert the ROM in the middle one, with the little notch pointing to the left (ie, away from the near edge of the case). Switch on the machine, and look for the '+' sign after the

'Acorn MOS' message. If that's there everything is alright, and you should turn off the computer again, and screw up the case.

MOS Plus may also be fitted to a Master cartridge. Follow the fitting instructions which accompany blank cartridges.

The Disc version

This is even easier. Just press SHIFT and BREAK and MOS Plus will load into Sideways RAM bank 7. To load into another RAM bank instead, type:

***SRLOAD MOSPLUS 8000 <bank number>**

where the bank number may be 4, 5, 6 or 7. You must then press CTRL-BREAK to initialise the ROM. The features of MOS Plus are then available in exactly the same ways as the ROM version. Note that the *BACKUP command must not be used with the 'S' option, if you are using the disc version of MOS Plus, as this corrupts Sideways RAM.

Enhanced Star Commands

MOS Plus provides enhanced versions of many of the Master 128 '*' commands. Here is a list of the improvements. There are no commands removed, and every command can be used as before if you like.

***APPEND <fsp>**

This command now accepts all 256 characters from the keyboard, rather than only the first 128 as before. To generate characters above 128 from the keyboard function keys must be used. For example teletext control codes can now be included in text files generated with *BUILD and *APPEND. This feature appears on the Master Compact.

***BACKUP <source> <dest> (<S>) (<P>) (<Y>)**

Makes a copy of <source> disc onto <dest>. Options:

- S Uses Sideways RAM only to make the backup
- P Pauses between reads and writes (Used to copy Side 0 of one disc to Side 2 of another using a single drive)
- Y Bypasses the Y/N check.

***BASIC <@><hex addr>**

This is extended so that the line tokeniser in BASIC can tokenise text from any address in memory, not just the contents of EDIT's buffer. For example a BASIC program can be written in VIEW, saved and then *LOADED at &1000. Issuing *BASIC @1000 would tokenise the file, ready to LIST or RUN

***BUILD <fsp>**

Like *APPEND, this now also accepts all 255 characters when creating files.

***EDIT <fsp>**

This now correctly accepts the filename immediately after the command, as in *ED.FILE, or with many intervening spaces, as in *EDIT FILE. Previously exactly one space had to be used.

***ROMS**

This now lists Sideways RAM banks as 'RAM', not 'ROM', and does not list duplicate ROMs, as implemented on the Master Compact.

***SHOW (<key number>)**

This now allows the key number to be omitted, and the contents of all programmable keys are shown.

***SRLOAD <fsp> <sram address> <bank> (<Q>) (<I>)**

The command now has an additional optional parameter, 'I', which, after the loading has taken place, copies the byte at &8006 in that bank into &2A1+bank number. If the file loaded was a Sideways ROM image, this has the effect of immediately initialising it so that commands can be issued. This is also found on the Master Compact.

***SRWRITE <addr> <addr> <sram addr> <bank> (<I>)**

Like *SRLOAD, this will copy a byte after the data transfer. Both commands cause havoc if the data is not a valid ROM image.

***STATUS (<item>)**

This command now lists the MOS Status setting in alphabetical order, as on the Master Compact.

***UNPLUG <bank> (<I>)**

The command now allows ROMs to be UNPLUGged immediately, not on the next CTRL-BREAK, by following the bank number with the letter 'I' (for immediate).

New Star Commands

In addition to the extensions to existing commands, MOS Plus also provides a small set of new commands, documented below.

***BLIST <fsp> (<LISTO format>)**

This command will take a BASIC program <fsp> and display it, on screen, exactly as it would appear with the LIST command. This is useful if you are,

say, in the middle of word processing, and you want to check a BASIC file. Without leaving what you are doing, you can check any listing. The command works to the full Archimedes BASIC specification, and can decode all BASIC tokens up to and including BASIC V. The <LISTO format> option will take any number from 0-31, being bit significant codes for listing formatting where, if set:

- b0 A space is inserted between the line number and line text.
- b1 Program structures (FOR...NEXT etc.) are indented
- b2 Split the line at a colon
- b3 Don't display line numbers
- b4 List keywords in lower case.

***CATALL (<drive>)**

This new command which only works in ADFS, will catalogue all the files on an ADFS disc, correctly indenting directory levels. Directories are marked by following the name with the letter 'D'.

***DRIVE (<drive>)**

This ADFS command is provided to simulate the DFS equivalent for software compatibility. *DRIVE 1 is the same as *DIR :1.\$

***EXALL (<drive>)**

This new command is like *CATALL, but performs a *EX command on all files. Directories are similarly indented.

***FIND (<:drive><afsp>) (<D>)**

This will find a file anywhere on an ADFS disc, and display the filename, and full directory path. It's very useful if you save a file on disc, but can't remember where. If the command is followed by 'D', the directory names which match the <afsp> are printed. Wild cards # and * are supported.

***FORMAT <drive> <S|M|L> (<Y>) (<N>)**

This built-in ADFS formatter will save lots of time, as you can now *FORMAT discs without loading the Welcome disc formatter. The parameters are

- S 40 track single sided disc
- M 80 track single sided disc
- L 80 track double sided disc

The command responds by asking you to confirm the instruction, by typing 'YES'. This can be bypassed by adding a 'Y' parameter to the commands. Verification after formatting is normally automatic, but if you are in a hurry, you can bypass this by adding a 'N' parameter. Note that if using both options, you must type YN not NY.

***SETTIME <date|time|date and time>**

This command sets the date and time, using the same syntax as TIMES\$ in BASIC. Of course this can be used from within any language. The syntax is

***SETTIME "Wed, 11 Nov 1987.11:11:00" or**

***SETTIME "Wed, 11 Nov 1987" or**

***SETTIME "11:11:00"**

***SRKILL <bank id>...**

This command will wipe (the first 16 bytes of) one or more Sideways ROM images from memory, giving a quickway of clearing out ROM images so that *SRDATA can be used. To clear out banks 4 and 7 only, the syntax would be:

***SRKILL W Z**

***VERIFY (<drive>)**

This will verify an ADFS disc. If no drive number is given, the currently mounted drive is verified.

Alarm Clock System

The Master 128 contains an Alarm Clock facility as part of the CMOS RAM chip, which is not used by the Operating System. MOS Plus makes use of the facility to provide a permanent alarm system, which can be set to ring every second, minute or hour, or at a specified time. Two volume settings are allowed.

To use the clock, a link on the main circuit board must be set. The link, LK4, is a jumper with two pins. The link is marked on the board, and is near the middle of the left hand edge of the main PCB, to the right of the left-hand co-processor connector. Therefore any co-processor present must be removed. The pins should be connected using a jumper link, obtainable from any electronics shop, or in case of difficulty direct from Dabs Press, free on receipt of an SAE. Some Master boards don't have the pins present, in which case you will have to join the two pads with a small wire link, and solder them together. Again, if in doubt, consult Dabs Press. If you're not sure what you're doing, don't attempt the job. **Dabs Press cannot be held responsible for any damage to machines.**

Turn your machine off to set the link, and turn it on again. To test the alarm, make sure MOS Plus is installed, and type ***ALARM S** The machine should start beeping every second. The commands available under the alarmsystem are:

- *ALARM** Prints alarm setting on screen. If the alarm is set to an illegal time, asterisks will be printed.
- *ALARM <time>** Sets the alarm. The time must be given in the format <HH:MM:SS>. The alarm will then ring at the specified time, for about 90 seconds. To terminate the sounds, type ***ALARM OFF** (permanently stopped) or press **BREAK** (stopped until the same time tomorrow).
- *ALARM S** Sounds the alarm every second. ***ALARM M** sounds the alarm every minute on the minute, and ***ALARM H** every hour on the hour.
- *ALARM ON** Enables the alarm.
- *ALARM OFF** Disables the alarm.
- *ALARM QUIET** Alarm will ring with low volume. This can be permanently set in CMOS RAM with ***CONFIGURE ALARM QUIET**.
- *ALARM LOUD** Alarm will ring with high volume. This can also be permanently set with ***CONFIGURE ALARM LOUD**.

Once the alarm is set, it will survive reset and power down.

When the alarm is sounded, a user event (no 9) is generated. This can be picked up by setting the event vector to point to your routine, and at the end of your routine, jump to the address represented by the old contents of the vector. Your routine, which should only work if the accumulator equals 9 on entry, will be called every time the alarm signal occurs. Additional event entry values are provided in the X register, as follows:

- 0 Alarm at specified time
- 1 Second time signal
- 2 Minute time signal
- 3 Hour time signal

If on exit from the event routine, you set the top bit of X, then the alarm bell will not sound.

Mouse Driver

Many software packages offer mouse facilities, but usually require that you have an AMX or similar ROM fitted. The code to convert mouse movements into co-ordinate data is contained in MOS Plus, allowing access to many mouse-driven packages. OSWORD &64 is used, and is called without entry parameters. On return the parameter block contains data as follows

XY,XY+1	X Co-ordinate (0-1279)
XY+2,XY+3	Y Co-ordinate (0-1023)
XY+4	X text co-ordinate
XY+5	Y text co-ordinate (only valid for 32 line modes)
XY+5	Buttons bit 4-0 always zero
	Bit 5 Right hand bottom 0=Pressed 1=Not pressed
	Bit 6 Middle button 0=Pressed 1=Not pressed
	Bit 7 Left hand button 0=Pressed 1=Not pressed

Bug Fixes

MOS Plus fixes two major bug in the Master Operating System. The first is the infamous *CLOSE bug where DFS files do not close with the correct length if *CLOSE or CLOSE#0 is used. This now works correctly. The second is that *EDIT now accepts any number of spaces between command and filename. Some improvements have been made. *REMOVE now fails if a second filename follows the first. This stops problems with using *REM. when you meant *REN. for *RENAME. The CMOS RAM R-Poweron reset now puts sensible numbers in the CMOS bytes, unlike the previous situation where all parameters were set to zero.

Help System

MOS Plus contains detailed help text on many Master features. When you type *HELP there are five options:

MOS+ 1.00

```

MOSPlus - For list of new commands
BASIC   - For list of BASIC keywords
View    - For help on View commands
Alarm   - For help on using the Alarm Clock
Terminal- For help on Terminal commands

```

*HELP VIEW can be followed by further commands, ie *HELP VIEW HIGHLIGHTS. Typing *HELP VIEW gives details.

© David Spencer 1987 •First published June 1987
 ISBN 1-870336-44-5 (ROM version),45-3 (5.25" disc),46-1 (3.5" disc)
 Manual by David Spencer and David Atherton
 All rights reserved.

Published by Dabs Press, 76 Gardner Road, Prestwich, Manchester M25 7HU
 England. Telephone 061-773 2413.

MOS

P L U S

Improvements and Fixes

This note details the bug fixes in versions of MOS Plus up to the current version 1.14, and to provide additional documentation for the extra features introduced in the latest version.

Version 1.09

*CLOSE now enables interrupts on return curing the lockup which occurs with OSCLI "CLOSE":A=GET, and the formatter now much more reliable.

Version 1.10

1. *CLOSE now keeps an error pointer.
2. Service &25 is no longer claimed allowing MOS+ to work with the 'CONVERT' ROM on the Welcome disc.
3. *VERIFY no longer crashes if an error is generated when running on a second processor.

Version 1.12

1. *FIND now gives a count of objects found at the end of the search, and now takes an optional search parameter.
2. *FIND takes an optional 'F' parameter [*FIND <list spec> (D) (F)] which lists full catalogue information, as in *EX, rather than just the filename. If the 'D' parameter is used, the 'F' must not precede it.
3. *FORMAT no longer corrupts main memory, and the bug which caused an intermittent crash on second processors when trying to format a write-protected disc finally removed.
4. *BACKUP with Sideways RAM now 5-10 times faster
5. ADFS errors now proper SCSI codes and not 1770 codes when Winchester in use.
6. DFS *BACKUP reports 'Write protected disc' and not 'Disc error 12'
7. MOS+ no longer claims a page of workspace in HAZEL

More overleaf ...

**DABS
PRESS**

Version 1.13

1. *FIND no longer fails if the required file is in a directory whose name is exactly ten letters long.
2. The ROM now works with MOS 3.23 and 3.26 (unreleased as at 25th Feb 88)
3. *FIND now correctly reselects screen memory at all times
4. *SRKILL now clears 127 (not 16) bytes of a bank, to avoid problems with long title strings.

Version 1.14

Quite a few new items are added in version 1.14. The mouse driver is now removable to avoid conflict with other mouse drivers such as the Nidd Valley Chaffeur. A start string can be set which will be executed rather like a !BOOT file on power-up and CTRL-BREAK. The screen colour can be set (if not starting in Mode 7, and the 50-byte CMOS data can be loaded and saved to disc, useful if you need to change the batteries. *ECHO works in the same way as the Archimedes command of the same name—you can perform VDU sequences wherever you can issue * commands ie, *ECHO |S|@|D|@|@|@ will create a blue screen. A printer specific command *LPRINT will send data to the printer only, so *LPRINT |[E will set bold type. (The |[command is equivalent to ESC)

1. *CONFIGURE MOUSE and *CONFIGURE NOMOUSE will enable or disable the mouse driver at next hard reset.
2. *CONFIGURE START (<string>). Set string to be pushed into keyboard buffer on hard resets. The string should be in GS format (ie, |G = CHR\$(7) etc.) and truncated to 20 characters. eg, *CONFIGURE START *CAT|M will make your Master catalogue a disc every time it is switched on, or CTRL-BREAK is pressed. To cancel any start string type *CONFIGURE START and *STATUS will show 'No Start'. With this command and a mains timer, you can start automatic tasks.
3. *CONFIGURE COLOUR <0-15>. Set background colour to be forced (by VDU 19) on each hard reset. Does not work if your startup mode is 7 or 135.
4. *LCONFIG <filename> will load fifty bytes of CMOS RAM. *SCONFIG <filename> will save fifty bytes of CMOS RAM
5. *ECHO <string> sends a GS format string to current output stream. *LPRINT <string> sends the GS format string to printer only (uses VDU 1)

The new *CONFIGURE commands all appear in the correct alphabetical place when listed with *STATUS. This is because MOS Plus entirely replaces the *STATUS command.